present invention. The method **400** commences at operation **402**, where the cache **200** receives a request for streaming data from the client system **120**. If it is determined, at operation **404**, that the request is for data in the MMS format (i.e, the client access is MMS), the method **400** proceeds to operation **406**. At operation **406**, the cache **200** determines if it has a previously stored copy of the requested data stream, retrieves the requested data from the origin server **110** if necessary, and returns a copy of the requested data stream to the client **120**. If it is determined, at operation **408**, that the client access is RTSP, the version of a media player associated with the request is identified. In some embodiments, an RTSP request is only permissible where the request is associated with a streaming data request originated with a media player running certain versions of a media player software. If the software version of the associated media player is identified as an impermissible version at operation **412**, a failure is reported to the requesting client at operation **414**.

[**0047**] Once it is determined that the request is the RTSP request and that it originated from an allowable software version of a media player, a lookup for the RTSP file session header is performed at operation **416**. The lookup is performed utilizing an RTSP lookup key (e.g., NetCache:rtsp://server_name:554/filename/header). If the RTSP session file header lookup returns a miss, then the streaming media cache **200** obtains SDP from the origin server, writes the necessary information to disk **270** at operation **418**, and performs a lookup for the RTSP stream object and data at operation **420**. The lookup is performed utilizing MMS lookup keys (e.g., NetCache:mms://server_name:1755/filename-validator/storage and NetCache:mms:mms/server_name:1755/filename-validator/object-num). If the RTSP session file header lookup returns a hit, then the method proceeds directly to operation **420**.

[**0048**] If the lookup for the RTSP stream object and data at operation **420** returns a miss (e.g., a miss for the entire stream or for a specific object on disk), the requested data is retrieved from the origin server **110** utilizing RTP at operation **422**. The requested data is then served to the requesting client at operation **424**.

[**0049**] In order to optimize disk utilization, media content is stored by the cache **200** in a unified format, which, in one embodiment, is the MMS format. At operation **426**, the requested data is manipulated to convert its format into a unified format, (e.g., the associated RTP headers are stripped and replaced with MMS headers), after which the data is stored on disc in the unified format.

[**0050**] If the lookup for the RTSP stream object and data at operation **420** returns a HIT (i.e., the requested data is being stored on cache **200** in the MMS format) on disk), the MMS headers of the requested data are replaced with the RTP headers at operation **428**. The method then proceeds to operation **424**.

[**0051**] It will be noted, that although the method **400** is described with reference to the MMS and RTSP protocols, the method may be utilized to achieve unified caching of media content and the ability to utilize one copy of the payload stored in the cache **200** to process client requests regardless of the associated access protocol.

[**0052**] An illustrative MMS/MMS-HTTP data header stored by the cache **200** on disk is illustrated in **FIG. 5A**. The header comprises hard coded data fields **502** and **504** that together indicate the beginning of the data packet, packet length field **506** (derived from RTP payload format header length), location ID field **508**, timestamp field **510** indicating real play time of the packet, flags field **512** indicating the type of an associated Advanced Systems Format (ASF) object, and packet length field **514** (derived from RTP payload format header and including MMS header).

[**0053**] An illustrative RTP data header that stored by the cache **200** on disk is illustrated in **FIG. 5B**. Flags **520** include version (identifying the version of RTP), padding (indicating whether the packet contains one or more additional padding octets at the end which are not part of the payload), extension bit (if the extension bit is set, the fixed header is followed by exactly one header extension), and other information. Field **522** stores M bit (a marker to allow significant events such as frame boundaries to be marked in the packet stream) and the payload type that identifies the format of the RTP payload and determines its interpretation by the application.

[**0054**] A sequence number in field **524** is initially assigned a random value and is incremented by 1 for each ASF data packet. Field **526** stores RTP timestamp that can be set from Send Time field of the AFS data header (or payload parsing information). Field **528** stores synchronization source (SSRC) identifier, which may be set to a random value. Fields **520-528** may be collectively referred to as the RTP header. Field **530** is a payload format header.

[**0055**] As illustrated in **FIG. 6**, when data is being served via the RTSP protocol, the MMS-HTTP header **602** is stripped from each packet and replaced with the RTP header **608** before sending data packets to the client.

[**0056**] In one illustrative embodiment of the present invention, the data packets are written to disk with MMS headers after the data is served to the client. As shown in **FIG. 7**, the fields of the MMS header **708** are derived from the RTP header **702** before the packet is written to disk.

[**0057**] Deriving the MMS LocationId **508** (also called packet id) from the RTP header **702** may be described as follows. While the sequence number **524** in an RTP header increments by 1 for every RTP packet, the MMS packet id **508** in a Master Boot Record (MBR) file may not follow the same pattern. The data from the various streams is multiplexed, or interleaved, so that samples with similar presentation times across all streams are close to one another in the data section of the ASF file. There is no information in the ASF file header, the RTP header, or the ASF data packet header that can be used to identify the actual MMS packet id in the ASF file. This packet id depends on the layout of all the streams in the file. When data is being streamed via RTSP, it is assumed that packet ids are incremented by 1. It is acceptable for packet ids to be non-incrementing across the objects stored on disk. It is also acceptable for packet ids to be duplicated in different objects.

[**0058**] In one illustrative embodiment, cache **200** uses packet id to prefetch data from the origin server **110** (in the case of MMS-HTTP, ASF offset may be used, which is calculated using the packet id). The prefetching is performed by packet ids. Because exact packet locations are not known to the cache **200**, the packet ids are estimated and generated by the cache **200**. Thus, the cache **200** may be requesting